Sécurité des RO

Partie II : Etude des attaques

Anas Abou El Kalam
anas.abouelkalam@enseeiht.fr

Définitions :

- **Vulnérabilité**

  faiblesse / faille : faute accidentelle ou intentionnelle introduite dans spécification, conception ou configuration du système

- **Attaque**

  Action malveillante qui tente d'exploiter une faiblesse dans le système et de violer un ou plusieurs besoins de sécurité

- **Intrusion**

  faute opérationnelle, externe, intentionnellement nuisible, résultant de l'exploitation d'une vulnérabilité dans le système

Définitions :

- **Menace**

  Violation potentielle d'une propriété de sécurité

- **Risque**

  Couple (menace, vulnérabilité)

Définitions :

- **DoS / DDoS : déni de service**

  attaque d'un serveur destinée à l'empêcher de remplir sa fonction.

  méthode classique : faire crouler le serveur sous une masse de requêtes généralement mal formées à dessein pour entraîner une réponse anormale et paralysante.

  L'attaque utilise très souvent une multitude de PC zombies travaillant de concert, infectés par des backdoors/chevaux de Troie et mobilisables à distance par un pirate.

  Il est aussi possible de bloquer à distance des routeurs en tirant parti de failles de leur software.

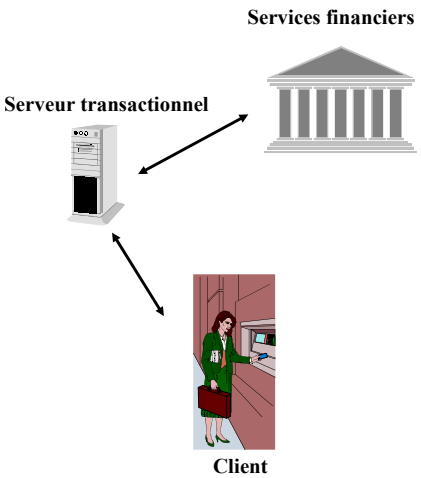
Plan chapitre 0

1. Définitions et principes généraux
2. Exemples de menaces de quelques SI
3. Critères généraux de la sécurité des SI

«Qui connaît l'autre et se connaît, en cent combats ne sera point défait;
qui ne connaît l'autre mais se connaît, sera vainqueur une fois sur deux;
qui ne connaît pas plus l'autre qu'il ne se connaît sera toujours défait.»

--L'art de la guerre - Sun Tzu

I. Carte à puces



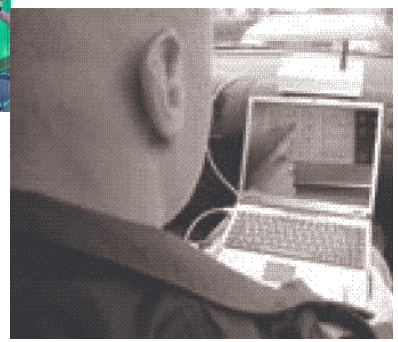
- Le client insère sa carte bancaire dans la fente et tape son NIP.
- L'écran demande le montant du retrait.
- Le client tape le montant du retrait et confirme.
- Le guichet crée une transaction et l'envoie au serveur transactionnel.
- Le serveur vérifie l'authenticité de la transaction et la relaie aux services financiers.
- La transaction est confirmée.
- Le guichet demande au client de retirer sa carte bancaire.
- Le guichet remet l'argent au client et imprime un relevé de transaction.

I. Fraude au distributeur

La fraude aux cartes peut s'opérer au niveau technologique : fausse goulotte pour lire la piste magnétique, caméra pour enregistrer la composition du code

I. Fraude au distributeur…

I. Fraude au distributeur…



Quelques références
- AFP 17/02/03 Nîmes
- AFP 09/04/03 Nice
- AFP 19/12/03 Meaux

Source panorama clusif 2003

I. Yescard et payements frauduleux

-La yescard est une carte à puce programmable qui permettait de faire des transactions d'achats sur quelques types d'automates de paiement électronique. La réflexion « théorique » a donné lieu à une fraude organisée avec un préjudice de plusieurs MF mais très localisée à quelques départements.

Chronologie
-Printemps 2001, création d'un groupe de « yescarder »
-Eté reportages presse et télé, exploitations « personnelles »
-Automne, mise en place de réseaux organisés notamment à proximité d'automates de distribution de carburant.

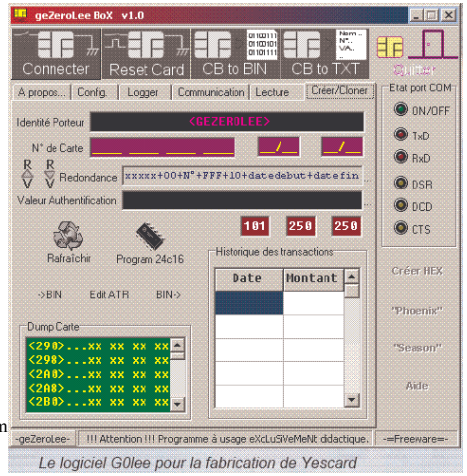
Source panorama clusif 2001

I. Yescard et payements frauduleux

Quelques détails
- En dessous d'un seuil de transaction d'achat, l'authentification de la carte et de son porteur sont fait en local.
- Seuls les automates (carburant, titre de transport, location vidéo, etc.) ont concernés

-Les DAB/GAB requièrent une demande d'autorisation en ligne.

- Les TPE chez les commerçants nécessiteraient la contre-façon visuelle de la carte ou une collusion.



Le logiciel G0lee pour la fabrication de Yescard

I. Yescard et payements frauduleux

Contexte
-Connaissance du principe dans le milieu professionnel
-Affaire judiciaire Serge Humpich vs GIE-CB
-Diffusion des clefs sur Usenet donne la décomposition du module opérationnel.

21359870359209100823950227049996287970510953418264
17406442524165008583957746445088405009430865999 =
11139543251488279879254901754770248440709228448 43*
19174817025245044393757862682308621806969341892 93.

-Migration EMV 5.1 et 5.2 depuis janvier 2002 avec durcissement du processus d'authentification et de non répudiation (clef 768bits)

2. Téléchargement illicites : les risques pour l'entreprise

Jugement TGI Marseille du 11/06/03 : Un employé de Lucent Technologies conçoit un site personnel dénonçant les abus (selon lui) de sa société Escota. Il met en ligne ce site depuis son poste de travail. Le tribunal de grande instance de Marseille condamne l'auteur de ce site mais aussi sa société en considérant que la faute a été commise dans l'exercice de ses fonctions (article 1384 du code civil)

Décision du conseil d'état du 15/10/03 : le conseil d'état confirme l'exclusion temporaire d'un adjoint technique de recherche. Cet employé avait utilisé l'adresse électronique de son directeur de laboratoire pour communiquer sur le site d'une secte. L'entreprise a été avertie de ce problème par un autre salarié et a a priori constaté le fait sur le site sans prendre connaissance du contenu des mails.

2. Téléchargement illicites : les risques pour l'entreprise

Cadre légal

La responsabilité pénale des employés est engagée en cas l'utilisation illicite de moyens informatiques de l'entreprise : droit l'auteur et des marques pour téléchargement de logiciels ›irates, documents audio ou films (mp3, DIVX, mpeg4,..), loi Godfrain (code pénal : 323.1, 323.2 et 323.3), pour les tentatives l'intrusion et altération d'un système

La responsabilité civile des entreprises peut aussi être établie si es tribunaux considèrent que l'employé en faute était « dans exercice de ses fonctions » en s'appuyant sur l'article 1384 du ;ode civil (ou responsabilité du commettant du fait du préposé)

http://www.legalis.net/jnet/2003/actualite_07_2003.htm
http://www.celog.fr/silex/tome1/chap_1-1.htm

4. La menace stratégique : Echelon

- **Un système espion : nom de code ÉCHELON**
  Accord secret UK / USA (1948) entre 5 organisations
  Espionnage militaire, de sécurité, civil, économique
- **Les moyens**
  Six bases d'écoute des télécommunications
  Réseau de satellites espions
  Puissance informatique : tri et filtrage

Satellite ground terminal at Etam, West Virginia, connecting Europe and the US via Intelsat IV

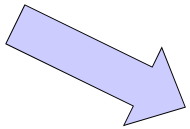
GCHQ constructed an identical "shadow" station in 1972 to intercept Intelsat messages for UKUSA

4. La menace stratégique : Echelon

- **Un exemple : la NSA**
  - La plus grande agence d'espionnage du monde
  - Créée en 52, plus de 20 000 agents
  - Budget annuel ~ 3 G€
  - Aide des armées et du National Reconnaissance Office
  - Concentration mathématiciens, linguistes, analystes
- **Clients**
  - Maison Blanche, Département d'État, CIA, FBI, Pentagone, DEA, Trésor, Département du Commerce
- **Missions**
  - Écouter, enregistrer, décoder, traduire, analyser les informations
  - Casser les codes de chiffrement
  - Élaborer des codes de chiffrement

Contexte stratégique : Quels agresseurs et quelles cibles

- Etats étrangers
- Espionnage économique
- Terrorisme politique et religieux
- crime organisé
- Hackers
- Utilisateurs



- Politiques
- Militaire
- Economique (industrielle, financière, commerciale)
- Scientifique

CLASSIFICATION DES ATTAQUES

Ecoute passive

- l'attaquant se contente d'accéder sans modification aux informations générées (e.g., sur un DD), transmises (sur canal de communication, e.g., MdP).

=> **sniffing, scanning, eavesdropping, wire tapping**

Interception

- l'attaquant modifie des informations transmises ou injecte des informations

=> rejeu, insertion, substitution, destruction, **TCP hijacking**

Déguisement

- l'attaquant tente de se faire passer pour quelqu'un d'autre

=>**IP spoofing**, **web page defacing, Fishing**

ryptanalyse

- **obtenir des informations secrètes à partir d'informations publiques**

=> à partir d'un message chiffré, retrouver le message en clair, retrouver la clé

CLASSIFICATION DES ATTAQUES

Déni de service

- l'attaquant vise à empêcher sa victime de continuer à délivrer le service (attaque de la dispo..)

=> spamming, flooding, smurfing

Bombe logique

ont aussi néfastes que les virus ou les vers et sont la cause de dégats similaires. La différence est que la bombe logique a besoin d'un détonateur pour s'activer,

C'est donc une fonction dévastatrice (**partie de programme qui reste dormante**) déclenchée à retardement ou par certaines conditions /

- date précise (vendredi 13)
- présence de certains utilisateurs, certains logiciels ou matériels
- après un certain nombre d'activations

=> destruction d'infos stockées : données, progs, infos de sécurité (ex: par formatage DD)
=> diffusion de fausses informations de diagnostic
=> dégâts matériels : usure anormale de périphériques, destruction d'écrans, de disquettes, l'imprimantes, etc.

CLASSIFICATION DES ATTAQUES

Porte dérobée (Backdoors)

- Accès cachés, moyen de contourner les mécanismes de sécurité ;
- faille du système de sécurité due à une faute de conception accidentelle ou intentionnelle (cheval de Troie en particulier)

- Ces passages secrets sont ménagés par les concepteurs de logiciels pour fournir des accès privilégiés pour les tests ou la maintenance.
  - Mais les pirates qui les découvrent peuvent déjouer tous les mécanismes de sécurité et rentrer dans le système.
- Parfois installée par un attaquant qui a réussi à s'introduire dans le système et qui veut se laisser un moyen de se réintroduire dans le système

exemple :

- août 1986, Lawrence Berkeley Laboratory : un intrus utilise le compte d'un utilisateur puis acquiert les privilèges systèmes : faille banale (mauvaise installation de Gnu-emacs)
- plutôt que de corriger la faille, il est décidé d'observer
- en 10 mois, l'intrus a tenté de pénétrer dans 430 ordis reliés à Internet, principalement militaires
- il a été possible de localiser et d'identifier l'intrus : un allemand de Hanovre, Markus Hess, lié au Chaos Computer Club et travaillant pour le KGB (condamné le 15 février 1990 à des peines de 2 ans de prison avec sursis et de 10 000 DM d'amende)

CLASSIFICATION DES ATTAQUES

Porte dérobée (Backdoors)

- Ex de backdoors sur certains logiciels (messagerie, utilitaires systèmes)
  - e.g., pour certaines commandes suivies d'arguments particuliers ou avec un mot de passe bien défini, le logiciel peut avoir un comportement différent
    - Permettre accès root à l'utilisateur, renvoyer un shell système à l'utilisateur, etc.
- Ces "trappes" sont inclues directement dans le code du logiciel.
- Certains développeurs sont soucieux de posséder un accès sur tous les systèmes utilisant leurs logiciels !!
  - e.g., Ken Thompson, l'un des pères d'UNIX, avoue avoir modifié l'application /bin/login en permettant l'accès direct au système par la saisie d'un mot de passe précompilé en dur. Thompson pouvait ainsi visiter tous les systèmes utilisant son application modifiée.
- Parfois, certains pirates diffusent des applications infestées de backdoors.

Protection ?

- MAC & Checksums & Signatures
- Veille : CERT, BugTraq, ...

télécharger ses applications sur le site du distributeur ou du programmeur.

- Utiliser des serveurs de téléchargement non liés à l'auteur de l'application peut se révéler dangereux.

CLASSIFICATION DES ATTAQUES

cheval de Troie

❖ programme effectuant une fonction illicite tout en donnant l'apparence d'effectuer une fonction légitime ;

- la fonction illicite peut être de divulguer ou d'altérer des informations, ou peut être une bombe logique

Exemple **AIDS**

- entre le 8 et le 12 décembre 1989, environ 10.000 disquettes "publicitaires" ont été envoyées à 7000 abonnés de PC Business World (GB) et à 3000 participants européens d'une conférence de l'OMS sur le SIDA en octobre 1988.
- l'enveloppe contenait : le texte d'une licence et le mode d'emploi d'installation
- objet prétendu du programme : évaluer le risque d'être atteint par le SIDA
- en fait, après 90 démarrages du PC, une bombe logique était lancée :
  - - chiffrement des noms de fichiers sur le disque
  - - édition d'un bon de commande pour payer la licence à l'ordre de PC-Cybor

Protection ?

- Antivirus
- MAC & Signatures
- Veille : CERT, BugTraq, ...
- Ne pas faire confiance aux prog non sûrs, Macros, ...

...

CLASSIFICATION DES ATTAQUES

RootKits

❖ programme permettant d'automatiser la dissimulation et l'effacement des traces d'un pirate sur une machine.

❖ Modifier commandes permettant d'administrer système, cacher ports ouverts par pirate...

❖ Utilisent le principe des backdoors

❖ ++ types : ls, ps, netstat ==>Modif lib ==> Modif comportement noyau, par le biais de modules chargés en mémoire

Protection ?

Les checksums.

- effectuer checksums à la fin d'une installation sur les différents fichiers comme ls, ps, stat ifconfig, etc. et sur les différentes bibliothèques partagées.
- Cette BD devrait être stockée sur CDROM ou autre support non réinscriptible.

2. Compiler les programmes vitaux en statique.

- disposerez d'une trousse de secours en cas d'infection par rootkits.
- pour cela, il faut disposer d'un OS permettant accéder sources progs vitaux...

3. Chkrootki tpermet de détecter la présence d'un rootkit (sous FreeBsd, libre).

4. Compilez noyau en statique. Vous éviterez ainsi chargement modules externes.

Classification attaques :

❖ Spyware

contraction de spy et software.

Logiciel espion qui collecte des données personnelles avant de les envoyer à un tiers,

e.g., Keylogger : transmettre les données saisies au clavier

❖ Spamming

'usage abusif d'un système messagerie destiné à exposer délibérément (et de manière répétée) les utilisateurs à des contenus non pertinents et non sollicités

CLASSIFICATION DES ATTAQUES

- **virus**

segment de programme qui, lorsqu'il s'exécute, se reproduit en s'adjoignant à un autre programme (du système ou d'application), et qui devient ainsi un cheval de Troie ;
Propriétés : infection, multiplication, fct nocive

=> éventuellement porteur d'une bombe logique
=> propagation par échange de support (disquettes) ou par réseau
peut simplement s'adjoindre à un autre programme et être donc exécuté uniquement orsque ce programme est exécuté
peut s'installer dans le secteur de boot d'une machine, d'une disquette ou dans la table des nterruptions : on parle de *virus résident*

macros-virus

- dans certains documents, des commandes peuvent être enregistrées comme des lonnées (formules de calcul dans les tableurs ou macros dans des traitements de texte)
- ces commandes peuvent être exécutées à l'ouverture du fichier
=> un virus peut s'insérer dans ces commandes
=> un logiciel comme Excel prévient de la présence de telles macros et demande 'autorisation pour les exécuter

virus de messagerie

- un cheval de Troie est inséré dans un document attaché à un courier électronique
- tout est fait pour que l'utilisateur exécute le document attaché (titre trompeur, ...)

CLASSIFICATION DES ATTAQUES

- **virus suite**

virus "*vendredi13*" (Israël, décembre 1987) :
+ virus pour PC,avec une bombe logique
+ détruit les programmes lorsqu'il s'exécute un vendredi 13

virus "*nimda*" (septembre 2001) :
+ virus pour PC, contenant un cheval de Troie;
+ ce virus se répand par fichiers attachés dans la messagerie
+ pas de sujet, document attaché "README.EXE"
+ le virus peut s'exécuter automatiquement à l'ouverture du mail sous Outlook
+ il provoque le partage en écriture du disque local
+ il tente de se propager par les dossiers partagés
+ il tente de saturer les serveurs de messageries et scane massivement le port 80 de ertains serveurs web pour dégrader leurs performances

virus "*Iloveyou*" (2000) :
- virus de messagerie électronique
- le sujet est "I love you"
- un document attaché LOVE-LETTER-FOR-YOU.TXT.vbs contient un virus
- détruit beaucoup de fichiers locaux en les remplaçant par sa propre copie, ssaie également de se propager par messagerie et par IRC.

CLASSIFICATION DES ATTAQUES

- **virus suite**

virus "*Mimail*" (octobre 2003) :
+ virus de messagerie électronique
+ le sujet est Re[2] : our private photos, dont l'emetteur est ames@<domaine_destinataire>
+ un document attaché PHOTOS.ZIP contient un virus qui s'il est exécuté, 'envoie à toutes les adresses emails trouvées sur la machine et attaque le site arkprofits.com

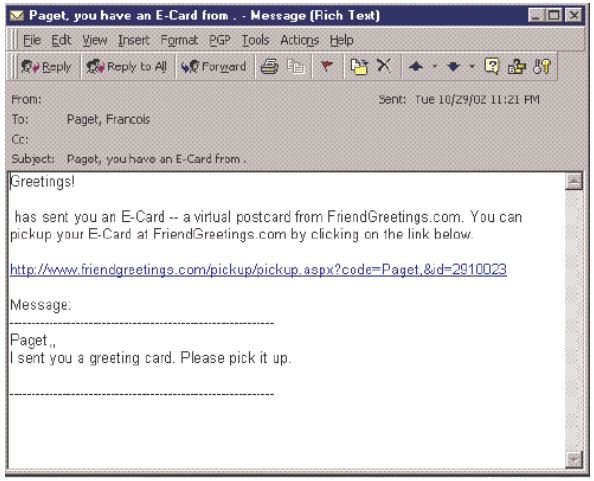
- En vrac : "*klez*", "*BugBear*", *"Sober"*

3. Les virus Internet, les vers : *Code Red*

- 17 Juillet 2001: le virus **CodeRed** commence une diffusion ultra rapide via Internet (250 000 systèmes infectés en moins de 9 heures).
  - cible les serveurs IIS de windows
  - utilise le protocole TCP/IP et le port 80.
  - défigure les pages web hébergées en y apposant la signature "Hacked by Chinese "
  - utilise un moteur de scan d'adresses IP puis s'auto-installe sur les systèmes vulnérables identifiés
  - entre le 1er et le 19 de chaque mois, le virus se propage, puis à partir du 20, il attaque (DoS) le site Web de la Maison Blanche

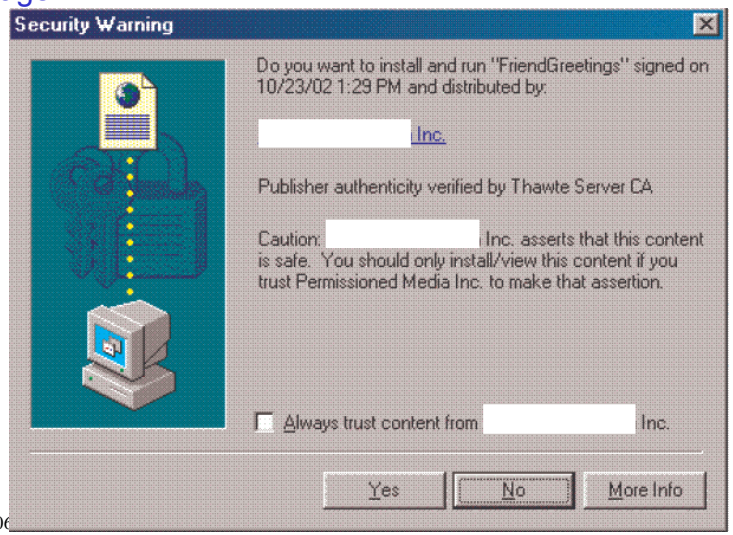
3. Exemple de mode de propagation des virus via email

24 Octobre 2002 : réception d'une carte virtuelle Friend Greeting application envoyée par un ami.


Paget, you have an E-Card from . - Message (Rich Text)
File Edit View Insert Format PGP Tools Actions Help
Reply Reply to All Forward
From:
Sent: Tue 10/29/02 11:21 PM
To: Paget, Francois
Cc:
Subject: Paget, you have an E-Card from .
Greetings!
has sent you an E-Card -- a virtual postcard from FriendGreetings.com. You can pickup your E-Card at FriendGreetings.com by clicking on the link below.
http://www.friendgreetings.com/pickup/pickup.aspx?code=Paget,&id=2910023
Message:
Paget ,,
I sent you a greeting card. Please pick it up.

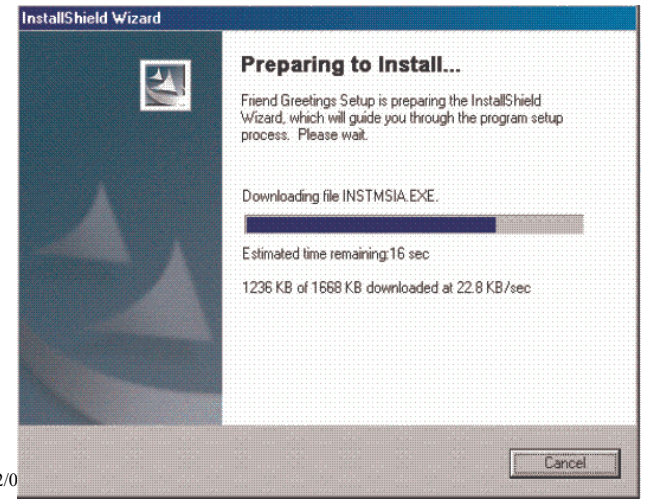
3. Exemple de mode de propagation des virus via email

Sans me méfier je clique sur le lien car j'ai envie de voir le message


Security Warning
Do you want to install and run "FriendGreetings" signed on 10/23/02 1:29 PM and distributed by:
Inc.
Publisher authenticity verified by Thawte Server CA
Caution: Inc. asserts that this content is safe. You should only install/view this content if you trust Permissioned Media Inc. to make that assertion.
Always trust content from Inc.
Yes
No
More Info

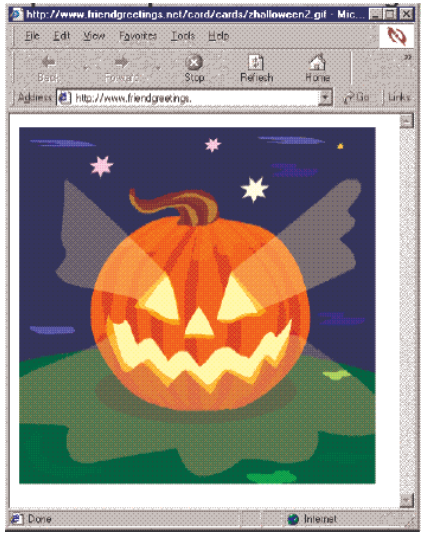
3. Exemple de mode de propagation des virus via email

Sans prêter trop attention à ce qui se passe, je répond oui à quelques questions et me voilà prêt à « relayer » des email


InstallShield Wizard
Preparing to Install...
Friend Greetings Setup is preparing the InstallShield Wizard, which will guide you through the program setup process. Please wait.
Downloading file INSTMSIA.EXE.
Estimated time remaining:16 sec
1236 KB of 1668 KB downloaded at 22.8 KB/sec
Cancel

3. Exemple de mode de propagation des virus via email

Me voilà récompensé !! Et rapidement inquiet en voyant le compteur de messages envoyés de mon poste !!!



CLASSIFICATION DES ATTAQUES

- **Vers**

programme autonome qui s'exécute, se propage et se reproduit à l'insu des utilisateurs normaux
=> éventuellement porteur d'une bombe logique
=> propagation par réseau

premier ver, Xerox, 1975 : programme "utile", composé de segments qui se copiaient sur les machines inactives du réseau (tests de machines, mesure de performances ...)

le *ver d'Internet* (2 novembre 1988) :
créé par Robert Morris Jr., étudiant, fils de Robert H. Morris ("*Chief Scientist*" du "*National Computer Security Center*" de la *National Security Agency*)
utilise 3 failles connues d'Unix dont l'attaque par dictionnaire des mots de passe
suite à une faute de conception, le ver se propage beaucoup plus rapidement que prévu par son auteur (condamné à 10 000 $ d'amende et 400 heures de travaux d'intérêt général)
=> immobilisation de 6000 à 60000 machines en une nuit
=> création du Computer Emergency Response Team (CERT) le 13 décembre

CLASSIFICATION DES ATTAQUES

- **Vers suite**

"*Blaster*" (connu également sous le nom de "*Lovsan*") :
- infecte les machines WindowsNT, Windows2000, Windows XP et Windows Server 2003
- exploite une faiblesse de DCOM (Distributed Component Object Model) qui utilise le port TCP/IP 135
=> suffit d'envoyer une requête particulièrement formattée pour provoquer une défaillance (*buffer overflow*)
- un email contenant un cheval de Troie a ensuite circulé en prétendant contenir un correctif pour le vers !
=> les correctifs de sécurité ne sont en général jamais envoyés par courrier !!

- modification de la base des registres
- génération d'une adresse IP et tentative d'infection de la machine correspondante
- création d'un processus qui écoute sur le port 1444 et qui permet d'exécuter à distance des commandes sur la machine infectée
- en fontion de certaines conditions (certains jours de certains mois), une attaque de type déni de service est lancée contre le site `windowsupdate.com` (SYN flooding)
- NB : le vers contient la chaîne de caractères :
"*I just want to say LOVE YOU SAN !*
billy gates why do you make this possible ? stop making money and fix your software !"

3. Les virus Internet, les vers : *NIMDA*

- Ver se propageant à l'aide du courrier électronique,

- exploite également 4 autres modes de propagation :
  - web
  - répertoires partagés
  - failles de serveur Microsoft IIS
  - échanges de fichiers

- Affecte particulièrement les utilisateurs de Microsoft Outlook sous Windows 95, 98, Millenium, NT4 et 2000.

3. Les virus Internet, les vers : *NIMDA*

- récupère @ présentes dans
  - carnets d'adresses de Microsoft Outlook et Eudora,
  - fichiers HTML présents sur le DD de la machine infectée.

- envoie à tous les destinataires un courrier dont
  - corps est vide,
  - sujet est aléatoire/long
  - pièce jointe nommée *Readme.exe* ou *Readme.eml*

- se propager à travers répertoires partagés des réseaux Microsoft Windows en infectant les fichiers exe s'y trouvant

- consultation de pages Web sur serveurs infectés peut entraîner une infection lorsqu'un utilisateur consulte ces pages (ie5)

- Exploite certaines failles de sécurité de IIS

3. Les virus Internet, les vers : *NIMDA*

Symptômes

- postes infectés possèdent sur leur disque fichiers :
  - README.EXE
  - README.EML
  - fichiers comportant l'extension .NWS
  - fichiers dont le nom est du type *mep*.tmp*, *mep*.tmp.exe*

Eradiquer

- déconnecter la machine infectée du réseau
- utiliser un antivirus récent / kit de désinfection de Symantec
- patch pour Microsoft Internet Explorer 5.01 et 5.5.

Définitions :

Ver	Virus
Autonome (n'utilisent pas nécessairement un fichier pour se propager), sur DD	parasites dissimulé dans fichiers ou dans code exécutable contenu dans secteur démarrage disque infectent en particulier les fichiers exe, ...
Arrive souvent par pièce jointe à un mail comprenant un code malicieux exécuté automatiquement par le logiciel de courrier électronique ou manuellement par l'utilisateur	Souvent par port réseau
ne se multiplie pas localement	se multiplie localement, s'autoduplique, se propage en infectant tour à tour les fichiers Effets : fichiers effacés, disque dur formaté, saturation des disques, modification du MBR

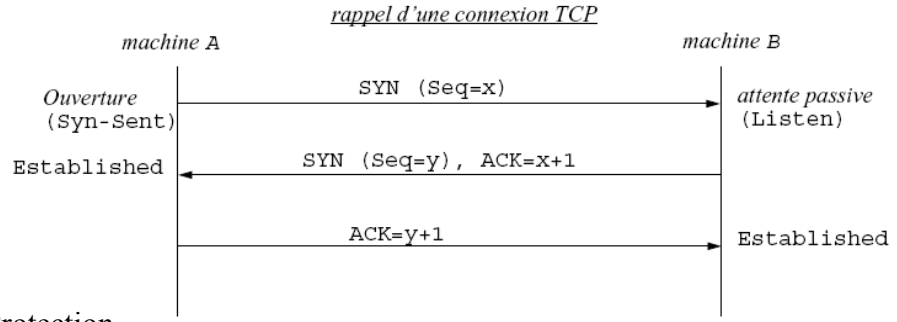
ATTAQUES RESEAU : Exemples

- IP n'est pas un protocole où la sécurité a été intégrée à la base
- IP est un protocole à datagramme
  - on ne peut faire confiance au champ adresse source
  => rien ne garantit que le paquet a bien été émis par la machine dont l'adresse IP est celle du champ source
- Les paquets sont routés de proche en proche
  => un routeur quelconque R peut envoyer un paquet à une machine B en se faisant passer pour une autre machine A
- Pas d'authentification des annonces de routage
  - des protocoles tels que RIP / OSPF permettent de mettre à jour dynamiquement les tables de routage par diffusion des routes
  => pas d'authentification des annonces donc déviation du trafic aisé !

ATTAQUES RESEAU : Exemples

SYN flooding (attaque de type *déni de service*)

Saturer le serveur en envoyant une grande quantité de paquets TCP avec le flag "SYN" armé, *sans répondre* aux "ACK" (connexion *semi-ouverte* sur la machine B)
=> consommation de ressources sur le serveur jusqu'à écroulement



Protection
e.g., avec **iptables** limitant demandes d'établissment de connexion TCP acceptées à 1/seconde

```
iptables -A FORWARD -p tcp --syn -m limit --limit 1/second -J ACCEPT
```

ATTAQUES RESEAU : Exemples

UDP flooding (attaque de type *déni de service*)

De la même manière que pour le SYN flooding, l'attaquant envoie un grand nombre de requêtes UDP sur une machine.
Le trafic UDP étant prioritaire sur le trafic TCP, ce type d'attaque peut vite troubler et saturer le trafic transitant sur le réseau.

e.g., *Chargen Denial of Service Attack*.

- Un pirate envoie une requête sur le port echo d'une machine A indiquant comme port source celui du port chargen d'une machine B.
- Le service *chargen* (initialement destiné à tester TCP/IP pour être sûr que paquets arrivent à destination sans altération) de la machine B renvoie un caractère sur le port echo de la machine A.
- Ensuite le service echo de A renvoie ce caractère sur chargen.
- chargen le reçoit, en ajoute un autre et les renvoie sur le port echo de A qui les renvoient à son tour sur chargen ...
- .. et cela continue jusqu'à la saturation de la bande passante.

Protection

Désactiver les services chargen et echo
Configurez firewall pour éviter le *Chargen Denial of Service Attack* en limitant traffic UDP

```
# iptables -A FORWARD -p udp --syn -m limit --limit 1/second -J ACCEPT
```


ATTAQUES RESEAU : Exemples

Packet Fragment (attaque de type *déni de service*)

envoyer par exemple des paquets ICMP de taille énorme
paquets de taille supérieure à 65536 octets peuvent provoquer un débordement du buffer de réception du datagramme : crash de la machine (*Ping of Death*)

- *Ping of Death* : envoyer paquet ICMP avec quantité données sup. à taille maximale d'un paquet IP . La pile peut s'avérer incapable de gérer cette exception et le reste du trafic.

sur un réseau de type Ethernet, le paquet est fragmenté et lors de la reconstruction du paquet, pas de vérif que la taille du dernier fragment n'est pas supérieure à la place dispo

Smurffing (attaque de type *déni de service*)

basée sur protocole ICMP. Lorsqu'on envoie ping à réseau en broadcast (ex 255.255.255.0), le paquet est envoyé à chacune des machines du réseau.
Un pirate envoie un ping en broadcast sur un réseau (A) avec une @IP source correspondant à celle de la machine cible (B). Le flux entre le port ping de la cible (B) et du réseau (A) sera multiplié par le nombre de machines sur le réseau (A).
Conduit à saturation bande passante du réseau (A) et du système de traitement de paquets de (B).

```
# iptables -A FORWARD -p icmp --icmp-type echo-request -m limit --limit 1/second -J ACCEPT
```

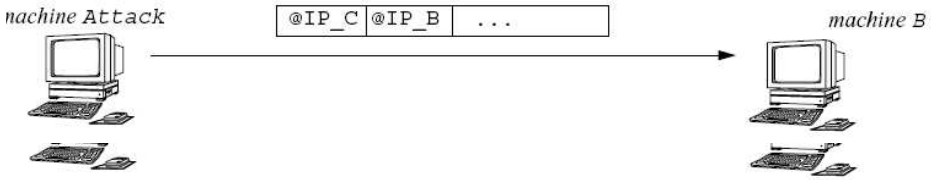
DDoS (attaque de type *déni de service*)

s'introduire sur plusieurs machines à partir desquelles l'attaquant va lancer une attaque sur une cible particulière
attaque en général orchestrée par un maître qui donne le signal

ATTAQUES RESEAU : Exemples

IP Spoofing (attaque de type *déguisement*)

la machine `Attack` envoie un datagramme IP en changeant l'adresse source et en la remplaçant par l'adresse source d'une autre machine
les messages de retour sont envoyés par la machine `B` à la machine `C`, donc `Attack` ne les reçoit pas
=> possibilité d'utiliser le champ `source routing` d'un datagramme IP qui permet de spécifier le chemin que va prendre un datagramme (*cf. transparent suivant*)



ATTAQUES RESEAU : Exemples

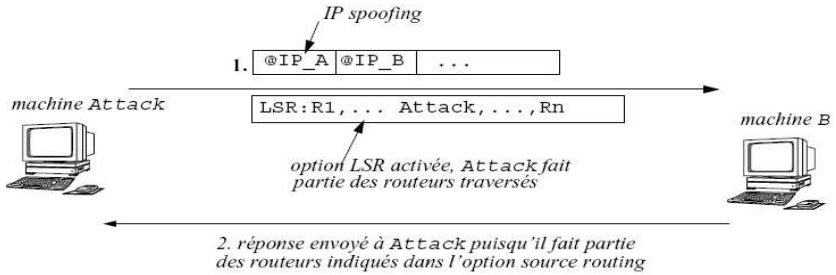
IP Spoofing par Source Routing

la machine `Attack` envoie un datagramme IP en changeant l'adresse source et en la remplaçant par l'adresse source d'une autre machine
messages retour sont envoyés par machine `B` à la machine `C`, donc `Attack` ne les reçoit pas
=> utiliser champ `source routing` d'un datagramme IP pour spécifier chemin ...

- + `source routing` impose la route exacte à suivre pour aller à destination (la liste des différents routeurs à traverser est insérée dans le datagramme)
- + `loose source routing` impose la traversée de certains routeurs pour aller à destination (d'autres routeurs peuvent également être traversés)

IP Spoofing par Source Routing (suite)

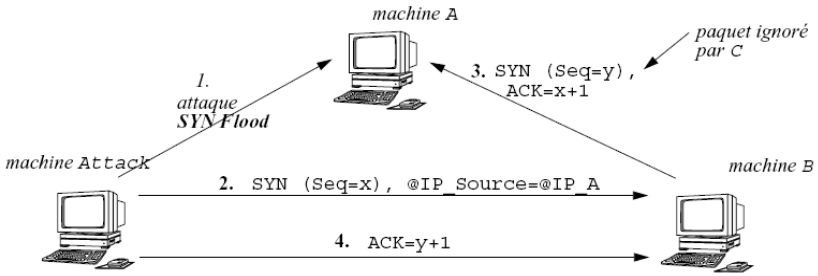
attaque : la machine `Attack` veut se faire passer pour la machine A auprès de B
. `Attack` choisit une machine A à laquelle B fait confiance
2. `Attack` crée un paquet IP en mettant comme adresse source celle de A
3. l'option `loose source routing` est positionnée
   • `Attack` fait partie des routeurs à traverser
4. le paquet retour utilise la même route que celle du paquet aller et passe donc par `Attack`
   • le paquet envoyé par `attaquant` est accepté par B puisqu'il semble venir de A, machine de confiance


IP spoofing
1. @IP_A @IP_B ...
machine Attack
LSR:R1,... Attack,...,Rn
machine B
option LSR activée, Attack fait partie des routeurs traversés
2. réponse envoyé à Attack puisqu'il fait partie des routeurs indiqués dans l'option source routing

IP Spoofing par prédiction du N° séquence TCP

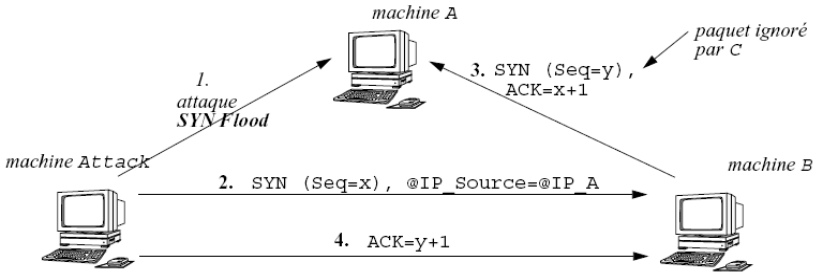
l'attribution des numéros de séquence TCP était faible dans l'implémentation originale
=> numéro de séquence est incrémenté d'une constante chaque seconde et de la moitié
le cette constante à chaque nouvelle connexion
si un intrus initie une connexion et visualise le numéro de séquence associé, il a de
ortes chances de deviner le numéro de séquence de la prochaine connexion

cette attaque s'utilise contre des services de type `rlogin` ou `rsh`, lorsqu'ils sont
onfigurés pour effectuer une authentification uniquement sur l'adresse IP source de
a machine cliente



IP Spoofing par prédiction du N° séquence TCP

`Attack` empêche A de terminer la connexion grâce à une attaque de type "SYN Flooding"
Si `Attack` est capable de prédire le numéro de séquence Y, il se fait donc passer pour A
. Après avoir déterminé une machine de confiance(ici la machine A), `Attack` met
ors service via un **SYN Flooding** par exemple (nécessaire pour qu'elle ne puisse
épondre aux paquets envoyés par B); ensuite, l'attaquant doit être capable de prédire numéros de
équence utilisés par la machine B
2. `Attack` initie une connexion TCP (`rsh` par exemple), en envoyant un paquet avec le flag
`SYN` à B et en se faisant passer pour A
3. B répond avec un paquet dont les flags `SYN` et `ACK` sont activés et dont le numéro de
équence vaut *y*
4. `Attack` renvoie à B un paquet contenant le flag TCP `ACK` avec le bon numéro
cquitement, ceci en se faisant toujours passer pour A



IP Spoofing par prédiction du N° séqce : Protection ?

Nmap invoqué avec l'option -O et -v vous fournit une indication sur la difficulté qu'aura le
irate à procéder à une attaque par IP spoofing contre votre serveur.

```
# nmap -O -v 192.168.1.4
```

```
tarting nmap V. 2.54BETA31 ( www.insecure.org/nmap/ )
Host (192.168.1.4) appears to be up ... good.
nitiating Connect() Scan against (192.168.1.4)
dding open port 111/tcp
.
dding open port 139/tcp
.
emote operating system guess : Linux 2.1.19 - 2.2.19
TCP Sequence Prediction : Class=random positive increments
Difficulty=4687481 (Good luck !)
PID Sequence Generation : Incremental
Nmap run completed -- 1 IP address (1 host up) scanned in 3 seconds
```

Si, par malchance, lors d'un scan, vous obtenez un nombre très bas avec un message du type Trivial Joke", cela signifie que votre système est très vulnérable à une attaque par IP-Spoofing.

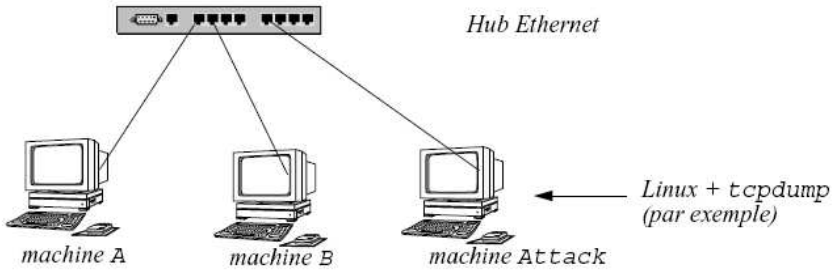
Autres moyens de protection

- Supprimer tous les services se basant sur l'authentification IP (rlogin, rsh).
- Certains modules comme rp_filter sous Linux permettent une défense contre ces attaques.
- Utilisation de tunnels

Sniffing (attaque de type écoute passive)

Exemple sur un LAN de type Ethernet

Possibilité d'observer tout le trafic entre la machine `A` et `B` sur la machine `Attack`
La notion de switch Ethernet et de commutation semble résoudre ce problème
=> pas vraiment, *cf. transparent sur le **sniffing** par utilisation de **spoofing ARP***



Outils "Snnifers »

Le sniffer place la carte réseau dans le mode transparent (promiscious),
- carte intercepte tous paquets sur le segment réseau, même ceux qui ne lui sont pas destinés

Tcpdump (tcpdump.org)
- affichent les données interceptées brutes ==> fichiers de log très volumineux

dsniff (www.packetstormsecurity.org)
- Récupére pwd & affiche directement à l'écran pwd, login, @ client, @ serveur

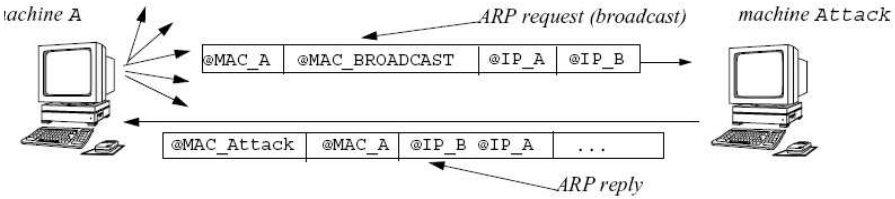
Ethereal / Wireshark(ethereal.com)
- affiche transactions ayant cours sur le réseau (notion de filtre)

Protection ?

Un sniffer est passif, il n'envoie aucun paquet, il ne fait qu'intercepter.
Mais la carte réseau étant en mode transparent, son comportement s'en trouve changé, son temps et sa façon de répondre à certains paquets sont modifiés.
On peut détecter la présence d'un sniffer grâce à ce changement de comportement.
- AntiSniff (sur win & linux) envoie paquets "tests" et en déduit si la carte est en mode transparent donc susceptible de sniffer.
- chiffrer transactions réseaux : "tunnels" (IPSec,VPN ...), SSL, ...

ARP spoofing (attaque de type déquisement)

- tout datagramme IP est encapsulé dans une trame Ethernet (avec @ source et @ destination)
le protocole ARP (*Address Resolution Protocol*) implémente le mécanisme de résolution d'une adresse IP en une adresse MAC Ethernet
si la machine source ne connaît pas l'adresse MAC de la machine destination, elle envoie une requete ARP en broadcast, attend la réponse et la mémorise



=> `Attack` répond en indiquant sa propre adresse MAC, ainsi elle intercepte les paquets à destination de `B`

Protection contre ARP spoofing

La solution la plus immédiate consiste à saisir manuellement sur chaque poste la table de toutes les adresses physiques présentes sur le réseau local.
- Si elle est immédiate, cette solution est quasiment inapplicable compte tenu du nombre d'hôtes connectés au réseau local.

Une solution correcte consiste à mettre en place un serveur DHCP avec une liste «fermée» de correspondance entre adresses physiques (MAC) et IP.
- La liste exhaustive des adresses physiques est centralisée sur le serveur DHCP.
- On peut ensuite configurer la journalisation du service pour que toute requête DHCP relative à une @ MAC inconnue génère un courrier vers l'aministrateur.

Enfin, On peut utiliser sous UNIX, un logiciel spécialisé : arpwatch
- Permet de surveiller tout le trafic ARP.

Les NIDS peuvent aussi détecter ce type d'attaques (Prelude-IDS, Snort, ...).

Sniffing par utilisation de ARP spoofing

- la machine `10.0.0.227` (machine pirate) veut recevoir le trafic destiné à la machine `10.0.0.1`, passerelle par défaut de la machine `10.0.0.171` (machine victime)

arpspoof –t victime passerelle

- *Attaquant : dire à la victime que mnt, nous (notre @MAC) appartient à l'IP de passerelle*

echo 1 > /proc/sys/net/ipv4/ip_forward

- *Permettre l'"IP forwarding" de manière à ce que le traffic passe par le host de l'attaquant*

```
root@victime [~] traceroute 10.0.0.1
traceroute to 10.0.0.1 (10.0.0.1), 30 hops max, 40 byte packets
 1  10.0.0.1  (10.0.0.1)  1.218 ms  1.061 ms  0.849 ms

root@victime [~] arp
Address          HWtype HWAddress           Flags Mask      Iface
10.0.0.1         ether   00:b0:c2:88:de:65   C               eth0
10.0.0.227       ether   00:00:86:35:c9:3f   C               eth0

root@pirate [~] arpspoof -t 10.0.0.171 10.0.0.1
 0:0:86:35:c9:3f 0:60:8:de:64:f0 0806 42: arp reply 10.0.0.1 is-at 0:0:86:35:c9:3f
0:0:86:35:c9:3f 0:60:8:de:64:f0 0806 42: arp reply 10.0.0.1 is-at 0:0:86:35:c9:3f

root@victime [~] arp
Address          HWtype  HWAddress           Flags Mask      Iface
10.0.0.1         ether   00:00:86:35:c9:3f   C               eth0
10.0.0.227       ether   00:00:86:35:c9:3f   C               eth0

root@victime [~] traceroute 10.0.0.1
traceroute to 10.0.0.1 (10.0.0.1), 30 hops max, 40 byte packets
1 10.0.0.227 (10.0.0.227) 1.712 ms 1.465 ms 1.501 ms
2 10.0.0.1 (10.0.0.1) 2.238 ms 2.121 ms 2.169 ms
```

paquets ARP empoisonnant le cache ARP de la machine 10.0.0.171 avec des ARP reply indiquant que l'adresse MAC associée à 10.0.0.1 est 00:00:86:35:c9:3f

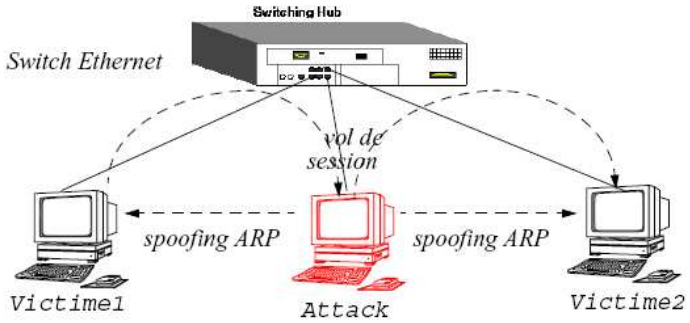
Pour vérifier que le trafic passe maintenant par la machine 10.0.0.227

Sniffing par utilisation de ARP spoofing

- `Attack` corrompt le cache ARP des machines `Victime1` et `Victime2` en envoyant des faux *ARP reply*

==> il envoie sa propre adresse MAC lorsque `Victime1`(resp. `Victime2`) essaie de connaître l'adresse MAC de `Victime2` (resp. `Victime1`)

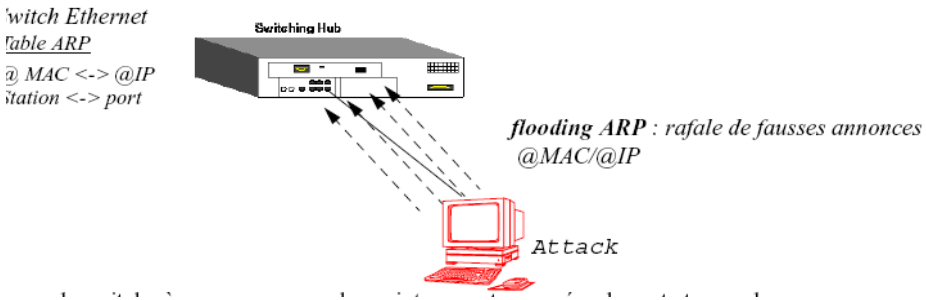
=> il intercepte ainsi la communication entre `Victime1` et `Victime2`



Sniffing par utilisation de ARP flooding

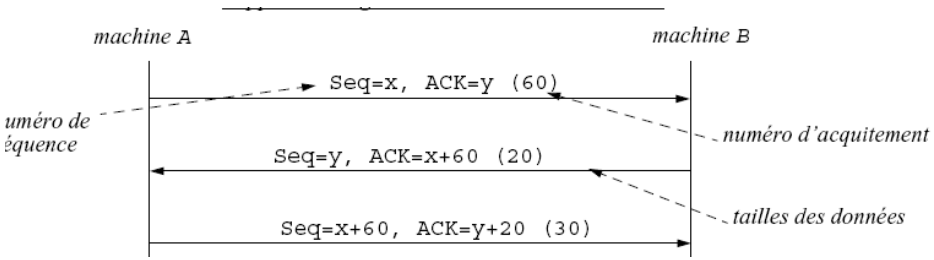
- le switch gère une correspondance interne entre numéro de port et une adresse Ethernet :
  - il n'envoie donc les paquets destinés à une adresse Ethernet particulière que sur le port correspondant
- l'attaque consiste à envoyer de multitudes de faux couples (@MAC/@IP) pour saturer les tables du switch

=> le switch passe alors en mode "hub" et envoie toutes les trames sur tous les ports



TCP session hijacking (attaque de type interception)

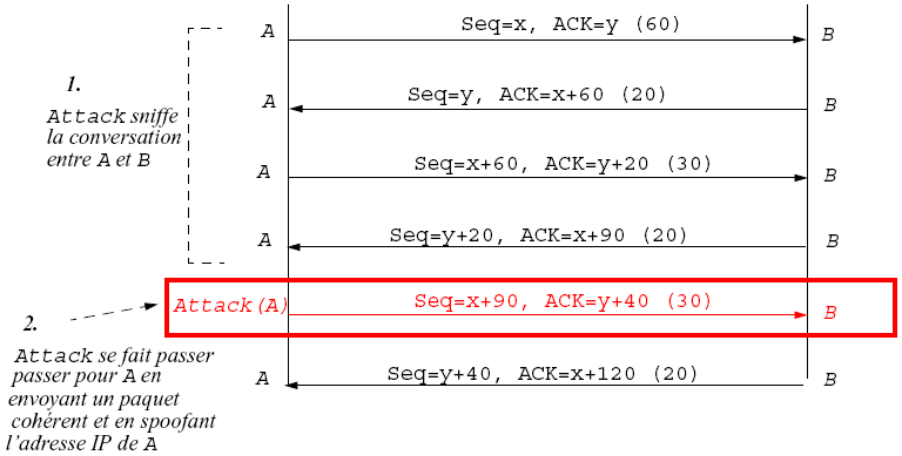
-Rappel : échange de données en connexion TCP



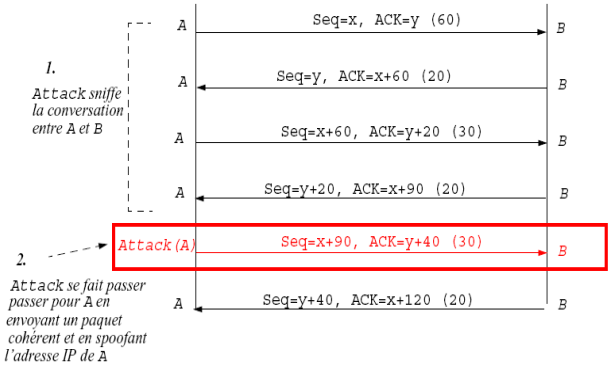
- l'attaque consiste, pour une machine `Attack`, à attendre la phase de connexion et à usurper l'identité de `A` pendant l'échange de données

=> ceci suppose que `Attack` est capable de sniffer les connexions entre `A` et `B`

TCP session hijacking (attaque de type interception)


1.
Attack sniffe la conversation entre A et B
Seq=x, ACK=y (60)
Seq=y, ACK=x+60 (20)
Seq=x+60, ACK=y+20 (30)
Seq=y+20, ACK=x+90 (20)
2.
Attack (A)
Seq=x+90, ACK=y+40 (30)
Attack se fait passer passer pour A en envoyant un paquet cohérent et en spoofant l'adresse IP de A
Seq=y+40, ACK=x+120 (20)

TCP session hijacking


1.
Attack sniffe la conversation entre A et B
Seq=x, ACK=y (60)
Seq=y, ACK=x+60 (20)
Seq=x+60, ACK=y+20 (30)
Seq=y+20, ACK=x+90 (20)
2.
Attack (A)
Seq=x+90, ACK=y+40 (30)
Attack se fait passer passer pour A en envoyant un paquet cohérent et en spoofant l'adresse IP de A
Seq=y+40, ACK=x+120 (20)

cette attaque permet à `Attack` de voler une session telnet établie entre A et B :
1. dans un 1er temps, `Attack` sniffe trafic entre A et B et attend fin de l'authentification
2. ensuite `Attack` forge un paquet avec comme adresse IP source celle de A et le numéro d'acquittement attendu par B, B accepte donc ce paquet ; pour cela il faut que `Attack` ait écouté le trafic pour pouvoir correctement envoyer le numéro d'acquitement et de séquence

=> la connexion entre A et B devient désynchronisée

- paquet injecté par Attack fait que paquet légitime envoyé par A n'est plus accepté par B à cause du numéro de séquence, de même pour les paquets envoyés par B à A

=> apparaît le problème du *Ack Storm* (A et B envoient des ACK en permanence)

DNS spoofing (déguisement)

Le protocole DNS (Domain Name System) a pour rôle de convertir un nom complet de machine (par ex. `java.enseeiht.fr`) en son adresse IP (`193.50.169.118`) et réciproquement; protocole basé sur le mécanisme client/serveur (un serveur DNS répond aux requêtes des clients)

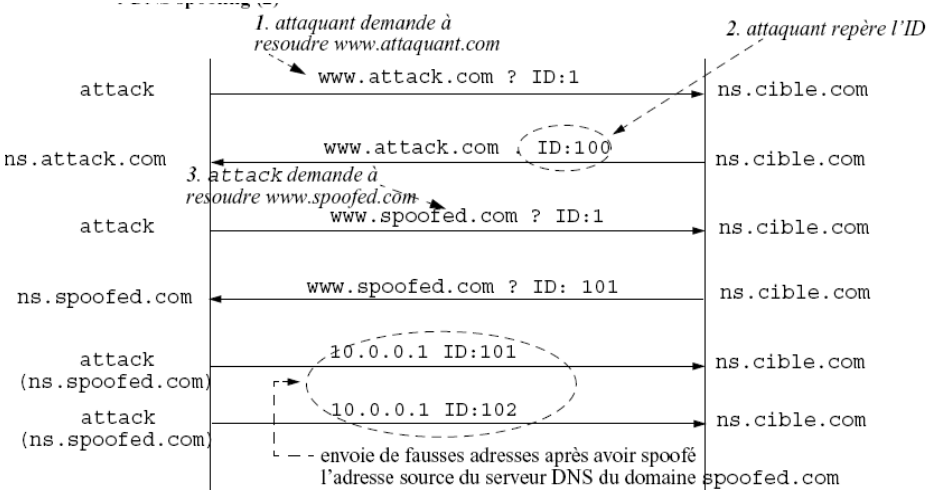
- L'attaque consiste à envoyer une fausse réponse à une requête DNS avant le serveur DNS

=> l'en-tête du protocole comporte un champ d'identification qui permet de faire correspondre les réponses aux demandes;

- la faiblesse provient de certains serveurs DNS qui génèrent des ID prévisibles

=> l'attaque nécessite que l'attaquant contrôle un serveur DNS (par ex `ns.attack.com`) ayant autorité sur le domaine `attack.com`

DNS spoofing (déguisement)



DNS spoofing (déguisement)

cette attaque nécessite que `attack` contrôle le serveur DNS `ns.attack.com`
et qu'il sait prédire les numéros de séquence DNS de `dns.cible.com`
1. `attack` envoie une requête DNS pour le nom `www.attack.com` au serveur DNS du domaine `cible.com`
2. le serveur DNS relaie la demande au DNS du domaine `attack.com`;
   ==> ainsi `attack` peut sniffer la requête pour récupérer l'`ID`
3. `attack` falsifie @ IP associée à un nom de machine, `www.spoofed.com`, et émet ensuite une requête de résolution pour `www.spoofed.com` vers `ns.cible.com`;
   ==> `ns.cible.com` relaie la requête en l'envoyant à `ns.spoofed.com`
4. `attack` envoie réponses DNS falsifiées à sa propre requête en se faisant passer pour `ns.spoofed.com` (plusieurs réponses pour avoir plus de chances de tomber sur le bon `ID`)
=> le cache DNS de `cible.com` est donc corrompu

Protection

- Configurez votre serveur DNS pour qu'il ne résolve directement que les noms de machine du réseau sur lequel il a autorité.
- Autorisez seulement machines internes à demander la résolution de noms de domaines distants.
- Mettez à jour ou changez les logiciels assurant le service DNS pour qu'ils vous protégent des attaques décrites précedemment.
- Voir http://www.linux-france.org/article/memo/dns/node16.html

Attaque RIP

diffusion dynamique de messages permettant de mettre à jour les tables de routage

aucune authentification à l'origine

=> possibilité pour un intrus de forger un paquet de façon à modifier les tables de routages d'un routeur

=> possibilité pour un intrus de recevoir certains paquets qui ne lui étaient pas destinés :

- il suffit qu'il envoie un message RIP forgé indiquant que lui-même est routeur pour certaines adresses de réseaux par exemple

Tiny Fragments attacks

un datagramme IP peut être fragmenté; dans ce cas, les filtres IP appliquent la même règle de filtrage à tous les fragments d'un paquet, la règle est déterminée au premier fragment

l'attaque consiste à fragmenter sur 2 paquets IP une demande de connexion TCP;

- le premier paquet IP de 68 octets (un paquet de 68 octets n'est jamais fragmenté à nouveau) ne contient comme données que les premiers octets de l'en-tête TCP
  - ports source et destination ainsi que numéro de séquence;
- les données du 2nd paquet IP renferment la demande de connexion TCP (flag `SYN` à 1)

=> si le fitrage est effectué sur les flags TCP (`SYN`, `ACK`), il échoue car le premier paquet n'en possède pas; le paquet peut alors être accepté
=> les paquets suivants passent également et à la destination, le paquet est reconstitué : la connexion est établie alors qu'elle aurait dû être refusée

Overlapping Fragments attacks

-- lorsqu'un paquet IP est fragmenté, le paquet complet est reconstitué lorsqu'il arrive à destination

- l'algorithme de réassemblage du paquet est décrit dans la RFC 791 :
  - il stipule que si 2 fragments ont eu une intersection non vide, c'est les données du dernier paquet reçu qui seront utilisées pour la reconstruction du paquet
- l'attaque consiste donc à
  - envoyer un premier fragment "inoffensif" (qui sera accepté par es filtres) et
  - ensuite de construire un deuxième fragment dont certaines données recouvrent celles du premier paquet (et qui contiennent donc une attaque)

Buffer overflow

- attaque qui n'utilise pas une faiblesse du réseau mais très utilisée pour attaquer les réseaux
  => consiste à exploiter des faiblesses des programmes serveurs qui s'occupent de services réseaux (server `ftp`, serveur `http`, ...)

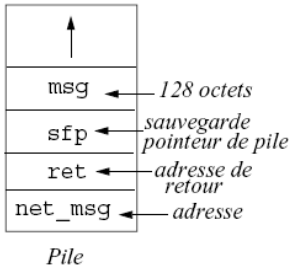
le principe : exploiter des programmes C pas assez rigoureux !

```
void copy_msg(char * net_msg)

msg[128];

strcpy(msg,net_msg);
return;
```

copie de `net_msg` dans `msg` sans vérifier la taille de `net_msg`



Buffer overflow

si `net_msg` est trés long , il peut écraser `msg` puis `sfp` puis l'adresse de retour de la fonction `copy_msg` (ce qui n'est pas normal !)

lorsque la fonction `copy_msg` se termine, on peut alors faire exécuter du code à une adresse choisie => justement celle de `msg`

- le code à exécuter est inséré dans net_msg qui est donc copié dans msg

```
Pile      DDDDDDDDEEEEEEEEEEEE EEEE FFFF FFFFFFFFFFFF
<---      89ABCDEF0123456789AB CDEF 0123 456789ABCDEF
          msg                   sfp  ret  net_msg
          [SSSSSSSSSSSSSSSSSS][SSSS][0xD8][adresse]
```

=> ici le code à exécuter est : `SSSSSSS........ S`
=> l'adresse de retour a été changée en `0xD8`, à l'endroit où a été placé le code à exécuter
=> l'art consiste donc à fabriquer `net_msg = SSSS.....SS0xD8`

Buffer overflow : example «Buffer overflow de wu-ftpd par MAIL_ADMIN»

Lorsque la fonctionnalité MAIL_ADMIN de wu-ftpd est activée, un attaquant distant pourrait provoquer un débordement de mémoire.

Le serveur wu-ftpd peut être compilé avec la fonctionnalité MAIL_ADMIN. Dans ce cas, un email est envoyé vers administrateur chaque fois qu'un fichier est uploadé sur le serveur.

```
From: wu-ftpd
Subject: New file uploaded: nom_fichier
```

Cependant, si la taille du nom de fichier avoisine les 32768 caractères, un buffer overflow se produit.

Un attaquant pourrait donc employer cette vulnérabilité dans le but de faire exécuter du code.

Buffer overflow : *example « Buffer overflow de ipcs »*

Le programme /usr/bin/ipcs fournit des informations sur l'usage des ressources IPC (Inter Process Communication).

- ipcs affiche le nombre de queues de message et de sémaphores créés ainsi que les segments de mémoire.

L'option "-C" indique la source d'informations à employer. En général, il s'agit d'un fichier core. Par exemple :

- **`ipcs -C /tmp/core`**

Cependant, ipcs ne vérifie pas la taille du nom de fichier !!

- Un attaquant local peut alors spécifier un nom de fichier core trop long dans le but de provoquer un buffer overflow.
  - e.g., permettre à un attaquant local d'accroître ses privilèges.

ATTAQUES RESEAU : Exemples

Attaque par format et buffer overflow de ntpd

Deux erreurs de programmation figurent dans le serveur `ntpd` (Network Time Protocol Daemon) et pourraient permettre à un attaquant distant d'accroître ses privilèges.

- `ntpd`: “daemon” qui met et maintient l'heure du système en synchronisation avec des serveurs de temps standards sur Internet
- `ntpd` s'exécute avec les privilèges du root
- But de l'attaque: obtenir accès root
  - Attaque : construire des réponses pour une requête avec des arguments “`readvar` ” longs
  - Il est facile de spoofer l'adresse source d'un serveur ntp

Attaques sur différents protocoles

Attaques contre Dynamic Host Configuration Protocol - DHCP

Attaque par épuisement de ressources

Un serveur DHCP possède un stock d'@ IP qu'il distribue aux différents clients

Ce stock est bien sûr limité. Il y aura seulement un nombre défini de clients pouvant disposer des différentes adresses IP

Scénario possible si DHCP est mal administré ; e.g., si les correspondances entre adresses MAC et IP se font dynamiquement à partir d'une plage d'@ IP vacantes

- Si un pirate génère un grand nombre de requêtes DHCP semblant venir d'un grand nombre de clients différents, le serveur épuisera vite son stock d'adresses.
- Les «vrais» clients ne pourront donc plus obtenir d'adresse IP
- Le trafic réseau sera paralysé.

Faux serveurs DHCP

Cette attaque vient en complément de la première.

- Si un pirate a réussi à saturer un serveur DHCP par épuisement de ressources, il peut très bien en activer un autre à la place.
  - Il pourra ainsi contrôler tout le trafic réseau.

DHCP : Protection

Limiter le service DHCP à une *liste «fermée»* de correspondances d'@ MAC et IP.

- Ainsi toute requête «étrangère» à cette liste est systématiquement rejetée.

Sous Windows,

- remplissez champs de l'option Réservations dans le prog de config du serveur DHCP

Sous Linux,

- éditez le fichier **/etc/dhcpd.conf** sur le serveur DHCP.
  - Par ex, pour client toto avec l'@ MAC 00:C0:34:45:56:67 à laquelle correspond l'@ 192.168.1.2, le routeur 192.168.1.1 et le serveur de noms 192.168.1.3.

```
host toto {
        hardware ethernet 00:C0:34:45:56:67;
        fixed-address 192.168.1.2;
        option routers 192.168.1.1;
        option domain-name-server 192.168.1.3; }
```

S'il est impossible d'établir une liste «fermée», *segmentez votre réseau* en sous-réseaux et attribuez-leur chacun un serveur DHCP.

- Ces serveurs seront indépendants les uns des autres.
- Les nouvelles versions du protocole DHCP permettent l'utilisation de *mécanismes d'authentification* plus stricts.
- Assurez vous que vos serveurs utilisent ces (*dernières) versions* de protocoles (RFC3118).

Attaques contre FTP

FTP

Transition des pwd en claire.

FTP anonyme

mauvaise gestion des droits d'accès peut s'avérer être une erreur fatale.
Laisser trop de répertoires en droit d'écriture et/ou d'exécution est plus que dangereux.
Le pirate pourrait y installer ou y exécuter des codes malveillants lui permettant d'accroître son pouvoir sur la machine.

Bouncing attack - Attaque par rebonds

Vise FTP anonymes
Utiliser serv FTP anonyme comme relais pour se connecter à d'autres serveurs FTP
Scénario

- Imaginons qu'un pirate se voit refuser l'accès par un serveur FTP dont l'accès est alloué à seulement un certain groupe d'adresses IP.
- Imaginons que le pirate ne fait pas partie de ce groupe, mais qu'un serveur FTP anonyme y appartienne.
- Le pirate peut se connecter sur le serveur FTP anonyme, utiliser les commandes assurant la connexion sur le serveur FTP protégé et y récupérer des fichiers

PROTECTION

•SFTP
•FTP anonyme seulement en cas de nécessité
•Gestion des droits (e.g., firewalls), Tuneling, ...

Attaques contre FTP

FTP

Transition des pwd en claire.

FTP anonyme

mauvaise gestion des droits d'accès peut s'avérer être une erreur fatale.
Laisser trop de répertoires en droit d'écriture et/ou d'exécution est plus que dangereux.
Le pirate pourrait y installer ou y exécuter des codes malveillants lui permettant d'accroître son pouvoir sur la machine.

Bouncing attack - Attaque par rebonds

Vise FTP anonymes
utiliser serv FTP anonyme comme relais pour se connecter à d'autres serveurs FTP
Scénario
Imaginons qu'un pirate se voit refuser l'accès par un serveur FTP dont l'accès est alloué à seulement un certain groupe d'adresses IP.
Imaginons que le pirate ne fait pas partie de ce groupe, mais qu'un serveur FTP anonyme y appartienne.
Le pirate peut se connecter sur le serveur FTP anonyme, utiliser les commandes assurant la connexion sur le serveur FTP protégé et y récupérer des fichiers

PROTECTION

SFTP
FTP anonyme seulement en cas de nécessité
Gestion des droits (e.g., firewalls), Tuneling, ...

SMTP : Relayage

Un serveur SMTP (port 25) sert à envoyer les mails sur le réseau local ou sur Internet.

Problème courant : le serveur SMTP peut servir de *relais* de mailing anonyme.

Un pirate peut très bien s'en servir pour envoyer des mails scabreux à travers Internet.

Un autre problème : commandes **EXPN** (@ d'un alias + liste récepteurs) et **VRFY**.

- Ces commandes sont sources de nombreuses informations pour le pirate.
- Il convient de les désactiver si le logiciel de messagerie le permet.

Comment s'en protéger ?

Appliquez des règles de firewalling assez strictes concernant le serveur SMTP

- Usage réservé exclusivement aux machines du réseau interne;
- tout ce qui sort de l'intérieur doit provenir d'une adresse interne; ....

Certains serveurs SMTP empêchent le relayage, vérifiez si votre serveur de messagerie supporte cette option.

DEMOS VIDEOS ATTAQUES

. . .

Maintenant qu'on connait notre ennemi

comment se protéger ?

. . .

SECURITE : LES MOYENS ?

- Identification, authentification
- Politiques d'autorisations et privilèges
- Gestion des droits et des privilèges
- Contrôles d'accès logiques et physiques
- Profils de protection, classes de fonctionnalités
- Évaluation, certification, accréditation, agrément, …
- Journalisation ("audit") des événements liés à la sécurité
- …